

RĪGAS TEHNISKĀ UNIVERSITĀTE

Datorzinātnes un informācijas tehnoloģijas fakultāte

4.praktiskais darbs

**Relaciju - objektu datu bāzes izveidošana**

Izstrādāja: Natans Šalamberidze

171RMC203

Pārbaudīja: lektors Jānis Eiduks

2023./2024. māc. Gads

Praktiskajā darbā jāiepazīstas ar relāciju – objektu datu bāzes tehnoloģiju:

1) objektu tabulu definēšanu, datu ievadi un vaicājumu veikšanu;

2) objektu tabulu sasaisti ar objektu saitēm (ne relāciju saitēm);

3) objektu metožu izveidošanu un izmantošanu vaicājumos.

Jāveic vaicājumu izpildes ātrdarbības salīdzināšana ar relāciju variantu.

Darba uzdevums.

1. Jāizveido divas loģiski saistītas relāciju tabulas (jādefinē un jāaizpilda ar datiem). Pirmās tabulas rakstu skaits >= 100 000, otrās tabulas rakstu skaits >= 1000. Var izmantot atbilstošas tabulas no iepriekšējiem darbiem. Starp tabulām jāvar definēt saiti viena ar daudziem (1 : M).

2. Tādas pašas tabulas jāizveido kā relāciju – objektu datu bāzes objektu tabulas (jādefinē un jāaizpilda ar datiem).

Realizējot objektu tabulas, tajās jāiekļauj elementi, lai varētu izveidot starp tām objektu saiti viena ar daudziem. Iespējami divi varianti. Papildus objektu tabulas ar kolekciju izmantošana, vai kolekcijas objektu iekļaušana lielākajā objektu tabulā. Obligāti darbā jāiekļauj datu bāzes (tabulas, sites, skati) grafiskais attēlojums.

3. Izdomāt 3 saliktus vaicājumus (vaicājumā vismaz 2 select izteiksmes un agregātfunkcijas).

4. Realizēt vaicājumus gan relāciju datu bāzes variantam, gan relāciju – objektu datu bāzes variantam.

5. Iegūt vaicājumu izpildes ātrdarbības novērtējumus, apkopot tos WORD tabulā un veikt, salīdzināšanu un izvērtēšanu.

6. Izmantojot abas objektu tabulas, izveidot objektu skatu, kura objektiem definēt divas MEMBER tipa objektu metodes.

7. Izdomāt divus vaicājumus objektu skatam, kuros tiek izsauktas definētās objektu metodes. Būtu ieteicams, lai ar šiem vaicājumiem un metožu izmantošanu, tiktu iegūti rezultāti, kurus ar objektu vaicājumiem ieguvāt iepriekš ar saliktiem vaicājumiem. Tad varētu salīdzināt ātrdarbības rādītājus un vaicājumu definēšanas ērtības (vienkāršāks pieraksts).

8. Secinājumi par darbu (apskatītās tehnoloģijas labās un sliktās īpašības, veikto piemēru rezultātu salīdzinājuma analīze).

**1. Jāizveido divas loģiski saistītas relāciju tabulas (jādefinē un jāaizpilda ar datiem). Pirmās tabulas rakstu skaits >= 100 000, otrās tabulas rakstu skaits >= 1000. Var izmantot atbilstošas tabulas no iepriekšējiem darbiem. Starp tabulām jāvar definēt saiti viena ar daudziem (1 : M).**

**2. Tādas pašas tabulas jāizveido kā relāciju – objektu datu bāzes objektu tabulas (jādefinē un jāaizpilda ar datiem).**

Realizējot objektu tabulas, tajās jāiekļauj elementi, lai varētu izveidot starp tām objektu saiti viena ar daudziem. Iespējami divi varianti. Papildus objektu tabulas ar kolekciju izmantošana, vai kolekcijas objektu iekļaušana lielākajā objektu tabulā. Obligāti darbā jāiekļauj datu bāzes (tabulas, sites, skati) grafiskais attēlojums.

**Vispirms veidojam relāciju tabulas:**

CREATE TABLE Autors (

autora\_id NUMBER PRIMARY KEY,

vards VARCHAR2(50),

uzvards VARCHAR2(50),

dzimis DATE,

valsts VARCHAR2(50)

);

CREATE TABLE Gramata (

gramata\_id NUMBER PRIMARY KEY,

nosaukums VARCHAR2(100),

izdevuma\_gads NUMBER,

autora\_id NUMBER,

FOREIGN KEY (autora\_id) REFERENCES Autors(autora\_id)

);

A screenshot of a computer

Description automatically generated

A diagram of a number and number

Description automatically generated

**Pēc tām veidojam objektu tipus un objektu-relāciju tabulas:**

CREATE OR REPLACE TYPE Autors\_Type AS OBJECT (

autora\_id NUMBER,

vards VARCHAR2(50),

uzvards VARCHAR2(50),

dzimis DATE,

valsts VARCHAR2(50)

);

/

CREATE OR REPLACE TYPE Gramata\_Type AS OBJECT (

gramata\_id NUMBER,

nosaukums VARCHAR2(100),

izdevuma\_gads NUMBER

);

/

CREATE OR REPLACE TYPE GramatuKolekcija\_Type AS TABLE OF Gramata\_Type;

/

CREATE OR REPLACE TYPE AutorsGramatas\_Type AS OBJECT (

autors Autors\_Type,

gramatas GramatuKolekcija\_Type

);

/

CREATE TABLE AutorsGramatas OF AutorsGramatas\_Type NESTED TABLE gramatas STORE AS GramatuKolekcijaStorage;

/

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a diagram

Description automatically generated

A yellow rectangular sign with black text

Description automatically generatedA yellow rectangle with black text

Description automatically generated

**Aizpildām abu tipu tabulas:**

DECLARE

v\_gramata\_counter NUMBER := 1;

v\_vards VARCHAR2(50);

v\_uzvards VARCHAR2(50);

v\_dzimis DATE;

v\_valsts VARCHAR2(50);

v\_nosaukums VARCHAR2(100);

v\_gads NUMBER;

v\_gramatu\_skaits NUMBER;

TYPE v\_gramatu\_kolekcija\_tips IS VARRAY(151) OF GRAMATA\_TYPE;

v\_gramatu\_kolekcija v\_gramatu\_kolekcija\_tips;

BEGIN

FOR i IN 1..1000 LOOP

v\_gramatu\_kolekcija := v\_gramatu\_kolekcija\_tips();

v\_vards := DBMS\_RANDOM.STRING('A', 10);

v\_uzvards := DBMS\_RANDOM.STRING('A', 10);

v\_dzimis := TO\_DATE('01-JAN-1880', 'DD-MON-YYYY') + DBMS\_RANDOM.VALUE(0, 365\*130);

v\_valsts := CASE ROUND(DBMS\_RANDOM.VALUE(1, 10))

WHEN 1 THEN 'Latvija'

WHEN 2 THEN 'Kanāda'

WHEN 3 THEN 'Vācija'

WHEN 4 THEN 'Francija'

WHEN 5 THEN 'Itālija'

WHEN 6 THEN 'Spānija'

WHEN 7 THEN 'Ķīna'

WHEN 8 THEN 'Austrālija'

WHEN 9 THEN 'Japāna'

ELSE 'Indija'

END;

INSERT INTO Autors (autora\_id, vards, uzvards, dzimis, valsts)

VALUES (

i,

v\_vards,

v\_uzvards,

v\_dzimis,

v\_valsts

);

v\_gramatu\_skaits := ROUND(DBMS\_RANDOM.VALUE(50, 151));

FOR j IN 1..v\_gramatu\_skaits LOOP

v\_nosaukums := DBMS\_RANDOM.STRING('A', 20);

v\_gads := ROUND(DBMS\_RANDOM.VALUE(1900, 2023));

INSERT INTO Gramata (gramata\_id, nosaukums, izdevuma\_gads, autora\_id)

VALUES (

v\_gramata\_counter,

v\_nosaukums,

v\_gads,

i

);

v\_gramatu\_kolekcija.EXTEND;

v\_gramatu\_kolekcija(j) := GRAMATA\_TYPE(v\_gramata\_counter, v\_nosaukums, v\_gads);

v\_gramata\_counter := v\_gramata\_counter + 1;

END LOOP;

INSERT INTO AutorsGramatas VALUES (

AutorsGramatas\_Type(

Autors\_Type(i, v\_vards, v\_uzvards, v\_dzimis, v\_valsts),

GramatuKolekcija\_Type(v\_gramatu\_kolekcija(1))

)

);

FOR k in 2..v\_gramatu\_kolekcija.COUNT LOOP

INSERT INTO TABLE (SELECT a.gramatas FROM AutorsGramatas a WHERE a.autors.autora\_id = i)

VALUES (v\_gramatu\_kolekcija(k));

END LOOP;

END LOOP;

COMMIT;

END;

/

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

**3. Izdomāt 3 saliktus vaicājumus (vaicājumā vismaz 2 select izteiksmes un agregātfunkcijas).**

**4. Realizēt vaicājumus gan relāciju datu bāzes variantam, gan relāciju – objektu datu bāzes variantam.**

Atrast vidējo, minimālo un maksimālo grāmatu skaitu, ko ir rakstījuši autori no katras valsts:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedIzvadīt autorus ar visvairāk grāmatām un to kopējo grāmatu skaitu:

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generatedAtrast piecus autorus ar augstāko vidējo grāmatu skaitu katrā desmitgadē:

A screenshot of a computer

Description automatically generated

**5. Iegūt vaicājumu izpildes ātrdarbības novērtējumus, apkopot tos WORD tabulā un veikt, salīdzināšanu un izvērtēšanu.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Vaicājums** | **Relāciju tabulas, laiks (s)** | **Objektu-relāciju tabula, laiks (s)** | **Attiecība (R/OR)** |
| 1 | 0.017 | 0.033 | ~0.52 |
| 2 | 0.019 | 0.025 | ~0.76 |
| 3 | 0.031 | 0.063 | ~0.48 |
| *Videjais* |  |  | *~0.59* |

Pēc iegūtam datiem var secināt, ka vaicājumi relāciju – objektu tabulā izpildās vairāk nekā 2 reizēs lēnāk, kas ļauj apgalvot, ka relāciju tabulu paņēmienam ir ievērojama priekšrocība. Bet tas nav taisnības visos gadījumos, kā arī visos aprakstītos vaicājumos, jo 2. vaicājums tika veikts gandrīz vienā laikā ar nelielu relāciju tabulu priekšrocību.

**6. Izmantojot abas objektu tabulas, izveidot objektu skatu, kura objektiem definēt divas MEMBER tipa objektu metodes.**

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

**7. Izdomāt divus vaicājumus objektu skatam, kuros tiek izsauktas definētās objektu metodes. Būtu ieteicams, lai ar šiem vaicājumiem un metožu izmantošanu, tiktu iegūti rezultāti, kurus ar objektu vaicājumiem ieguvāt iepriekš ar saliktiem vaicājumiem. Tad varētu salīdzināt ātrdarbības rādītājus un vaicājumu definēšanas ērtības (vienkāršāks pieraksts).**

**Atrast vidējo, minimālo un maksimālo grāmatu skaitu, ko ir rakstījuši autori no katras valsts:**

ALTER SYSTEM FLUSH SHARED\_POOL;

SET TIMING ON;

SELECT a.autors.valsts,

AVG((SELECT COUNT(\*) FROM TABLE(a.gramatas))) AS AvgBooksPerAuthor,

MIN((SELECT COUNT(\*) FROM TABLE(a.gramatas))) AS MinBooksPerAuthor,

MAX((SELECT COUNT(\*) FROM TABLE(a.gramatas))) AS MaxBooksPerAuthor

FROM AutorsGramatas a

GROUP BY a.autors.valsts

ORDER BY a.Autors.valsts;

/

SELECT

agv.autors.valsts,

AVG(total\_books),

MIN(total\_books),

MAX(total\_books)

FROM AutorsGramatas\_View agv

GROUP BY agv.autors.valsts

ORDER BY agv.autors.valsts;

SET TIMING OFF;

A screenshot of a computer

Description automatically generated

**Izvadīt autorus ar visvairāk grāmatām un to kopējo grāmatu skaitu:**

ALTER SYSTEM FLUSH SHARED\_POOL;

SET TIMING ON;

SELECT a.autors.autora\_id, a.autors.vards, a.autors.uzvards, a.autors.valsts, SUM((SELECT COUNT(\*) FROM TABLE(a.gramatas))) AS TotalBookCount

FROM AutorsGramatas a

GROUP BY a.autors.autora\_id, a.autors.vards, a.autors.uzvards, a.autors.valsts

ORDER BY TotalBookCount DESC

FETCH FIRST 5 ROWS ONLY

/

SET TIMING OFF;

ALTER SYSTEM FLUSH SHARED\_POOL;

SET TIMING ON;

SELECT agv.autors.autora\_id, agv.autors.vards, agv.autors.uzvards, agv.autors.valsts, SUM(total\_books) AS TotalBookCount

FROM AutorsGramatas\_View agv

GROUP BY agv.autors.autora\_id, agv.autors.vards, agv.autors.uzvards, agv.autors.valsts

ORDER BY TotalBookCount DESC

FETCH FIRST 5 ROWS ONLY

/

SET TIMING OFF;

A white paper with black text

Description automatically generated

**Atrast piecus autorus ar augstāko vidējo grāmatu skaitu katrā desmitgadē:**

ALTER SYSTEM FLUSH SHARED\_POOL;

SET TIMING ON;

SELECT ag.Autors.autora\_id,

ag.Autors.vards,

ag.Autors.uzvards,

AVG(BookCount) AS AvgBooksPerDecade

FROM AutorsGramatas ag

JOIN

(SELECT a.autors.autora\_id as autora\_id,

SUBSTR(g.izdevuma\_gads, 1, 3) || '0s' AS publication\_decade,

COUNT(\*) AS BookCount

FROM AutorsGramatas a,

TABLE(a.gramatas) g

GROUP BY a.autors.autora\_id, SUBSTR(g.izdevuma\_gads, 1, 3) || '0s') b

ON ag.Autors.autora\_id = b.autora\_id

GROUP BY ag.Autors.autora\_id, ag.Autors.vards, ag.Autors.uzvards

ORDER BY AvgBooksPerDecade DESC

FETCH FIRST 5 ROWS ONLY;

/

SET TIMING OFF;

ALTER SYSTEM FLUSH SHARED\_POOL;

SET TIMING ON;

SELECT agv.Autors.autora\_id,

agv.Autors.vards,

agv.Autors.uzvards,

agv.avg\_books\_per\_decade

FROM AutorsGramatas\_View agv

ORDER BY agv.avg\_books\_per\_decade DESC

FETCH FIRST 5 ROWS ONLY;

/

SET TIMING OFF;

A white background with black text

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| **Vaicājums** | **Objektu-relāciju tabulas, laiks (s)** | **Objektu skats ar metodēm, laiks (s)** | **Attiecība (Tab/Skats)** |
| 1 | 0.031 | 0.366 | ~0.08 |
| 2 | 0.023 | 0.195 | ~0.12 |
| 3 | 0.066 | 0.23 | ~0.29 |
| *Videjais* |  |  | *~0.16* |

Pēc iegūtam datiem var secināt, ka vaicājumi relāciju – objektu tabulā izpildās vairāk nekā 6 reizēs lēnāk, kas ļauj apgalvot, ka relāciju tabulu paņēmienam ir ievērojama priekšrocība. Bet tas ievērojami atvieglina vaicājumu veidošanu, kas īpaši labi redzams 3. vaicājumā, kur ar objektu metodes veidošanas iznāca izvairīties no apakšvaicājuma ar grupēšanu ar agregātfunkciju un no grupēšanas ar agregātfunkciju pati vaicājumā.

**Secinājumi par darbu (apskatītās tehnoloģijas labās un sliktās īpašības, veikto piemēru rezultātu salīdzinājuma analīze).**

Tika izpildīti visi dotie uzdevumi: tika izveidoti vajadzīgie datu tipi, divi objektu Member metodes, relāciju tabulas un relāciju-objektu tabulas, tabulas tika aizpildīti ar nejaušiem datiem ar uzrakstītas programmas palīdzību, tika veikti 3 dažādi vaicājumu relāciju tabulām, objektu-relāciju tabulai un objektu skatām ar metodēm, arī tika veikti vaicājumu izpildēs ātruma novērojumi un vērtējumi. Pēc iegūtam datiem var secināt, ka vaicājumi relāciju – objektu tabulā izpildās vairāk nekā 2 reizēs lēnāk, savukārt, vaicājumi objektu skatā ar metodēm izpildās 6 reizēs lēnāk nekā relāciju – objektu tabulā *(12 reizēs lēnāk nekā relāciju tabulās)*, kas ļauj apgalvot, ka relāciju tabulu paņēmienam ir ievērojama priekšrocība. Savukārt objektu metodes veidošana un to izmantošana skatā ievērojami atvieglina vaicājumu veidošanu, kas īpaši labi redzams 3. vaicājumā, kur ar objektu metodes veidošanas iznāca izvairīties no apakšvaicājuma ar grupēšanu ar agregātfunkciju un no grupēšanas ar agregātfunkciju pati vaicājumā.